Modular Development of STEP Application Protocols

ISO TC184/SC4/WG10
Julian Fowler, PDT Solutions (UK)

Overview

- Three related papers/presentations
  - Modular APs ⇨ WG10
  - Data Architecture & Common Resources ⇨ WG10, WG12
  - Process Improvement, Documentation ⇨ Quality Committee
- Intended to encourage discussion across WGs, development of concrete proposals

Current situation

- STEP APs are perceived as being:
  - difficult & expensive to develop
  - of variable quality
  - difficult to understand and review
  - difficult - and possibly ambiguous to implement
- Proliferation of APs is confusing
- We (SC4) need to do something about this!

QC/Process Improvement Findings

- Surveyed "customers" for APs:
  - end users
  - implementors
- Customer perceptions:
  - End user to understand the information management requirements
  - Implementor to develop products for users

QC/Process Improvement Findings

- **End users want:**
  - Scope
  - Definitions
  - Information requirements
  - ARM diagrams
  - Conformance classes
  - Implementation formats
  - Standard data/classes
  - *Application domain details*
  - *Usage scenarios*

- **Implementors want:**
  - AIM long form (with English explanation of constructs & how they compare to the "real world")
  - Mapping table
  - Abstract Test Cases
  - *Implementation guidelines*

- **Do <u>not</u> want:**
  - mapping table

QC/Process Improvement Recommendations

- Restructure 200/300 series parts to address end user and implementor needs separately
- Documentation changes
- Publish Integrated Resources on the Web
- Publish AP usages of the IRs

Analysis

- What customers *need* may not be what customers *want*
- Common thread (QC, I-ARMS, …)
  - <u>one</u> data model in an AP (not two)
  - a data model that is understandable and unambiguous
- Shift the cost/benefit basis for interpretation

Analysis

- AP projects invest in *analysis* and *modelling* directed towards the ARM
  - different analysis approach from that of STEP
- Result:
  - ARM and AIM are two different *solutions* to the application area's (implicitly stated) requirements
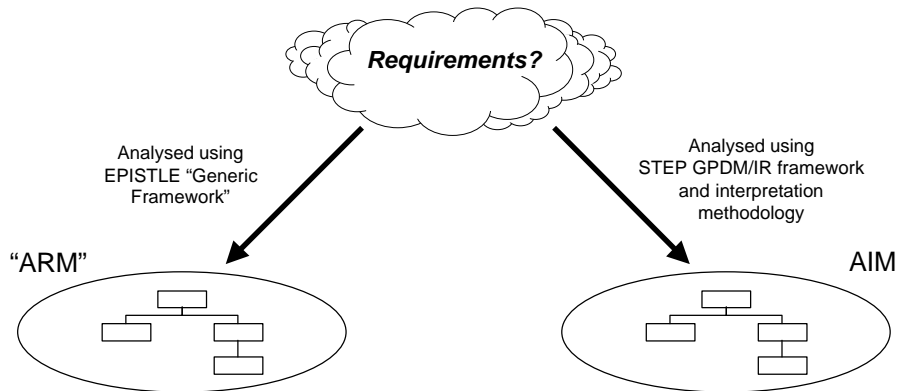
Extreme example: AP221

Requirements?

Analysed using
EPISTLE "Generic
Framework"

Analysed using
STEP GPDM/IR framework
and interpretation
methodology

"ARM"

AIM

What if …

- … an AP used the STEP GPDM framework to do *one* analysis of its requirements
  - underlying principles to guide the analysis
  - reusable data elements
  - emphasise overlaps and commonalities
- What would an AP look like?

GPDM analysis principles

- What application areas are to be supported?
- What kinds of things are we interested in, how are they identified?
- What are the important relationships?
- What aggregations of information are useful?
- What properties are we interested in?
- How are those properties described?
- ...

GPDM issues

- GPDM is not visible in STEP
  - some STEP participants are not aware of its existence!
  - confusion between GPDM and Part 41/43
- Existing documentation (Danner, Yang, Sanford, etc.) needs updating and refinement
- WG10 discussion (Chester) on metamodels

Possible "new look" for APs

- Clauses 1-3 unchanged
  - ISO requirements
- Clause 4: overview of requirements
  - "data planning model" level, with GPDM+ framework for analysis
- Clause 5: *the* data model
  - based on a richer set of resources
  - less specialisation/constraint that current AIMs

Emergence of modules

- More "top down" approach
  - Modules in the context of GPDM+
  - Emergence of basic patterns (cf. 100's, AICs)
- Subject area basis
  - UOFs, building blocks, AICs, conformance classes
- Modules + rules for combining modules

Example: AP221, AP227

AP221

- ✓ Identification of plant items
- ✓ Classification of plant items
  - use of common class library
- ✓ Composition of plant items
- ✓ Connection of plant items
- ✓ Physical and functional viewpoints
- • Properties (any)
- ✓ Description of properties (numeric, textual)
- • Schematic presentation
- • Life-cycle perspective

AP227

- ✓ Identification of plant items
- ✓ Classification of plant items
  - use of common class library
- ✓ Composition of plant items
- ✓ Connection of plant items
- ✓ Physical and functional viewpoints
- • Properties (limited)
- ✓ Description of properties (numeric, textual)
- • Shape representation (3D)
- • Shape presentation
- • "Snapshot" perspective

The basic building blocks

- • Context (meta data)
- • Identification
- • Classification
- • Definitions
  - design, manufacturing, functional, physical, …
- • Versioning
- • Composition, assembly
- • Connection, connectivity
- • Approvals, contracts, …
- • External references (documents, etc.)
- • Properties/characteristics
- • Data elements that describe properties
  - alphanumeric, geometric, parametric, …

Building an AP

- Requirements analysis using "GPDM+" framework
- Use of modules
- Standard/reference data elements
  - cf. AP221, POSC/CAESAR
  - existing mapping table constraints

Example: AP221, AP227

- AP221+
  - plant context, identification, classification, versioning, composition, configuration control, change management, connectivity, physical and functional aspects, properties, *2D schematic presentation*
- AP227+
  - plant context, identification, classification, versioning, composition, configuration control, change management, connectivity, physical and functional aspects, properties, *shape*, *3D shape representation*

Conclusions

- Evolution of the STEP AP methodology does permit a more modular approach
- We can identify a base set of modules quickly
  - GPDM+ framework
  - Experience from AP projects
  - Close fit with AP Interoperability results *and* WG10 work on SC4 data architecture

---

Issues

- Will require revisions to the Integrated Resources:
  - quality improvement
  - alignment with modules
  - inclusion of non-ISO 10303 elements (SC4 common resources)
- Legacy of existing APs
- Where do we start?
  - need a "testbed" to validate approach